DebugStringView Aplication

Introduction

It's an application that lets you monitor debug output on your local system. It's free and will display all the strings generated using the Windows OutputDebugString API.

Index

Installation and Use

The application does not require installation; simply unzip the file to the desired location and run it. Once started, it will begin capturing messages.

The application is distributed under the GNU GPL 3.0 license. You may use, distribute, and modify it as long as the same license is preserved. I would appreciate being informed of any modification or improvement.
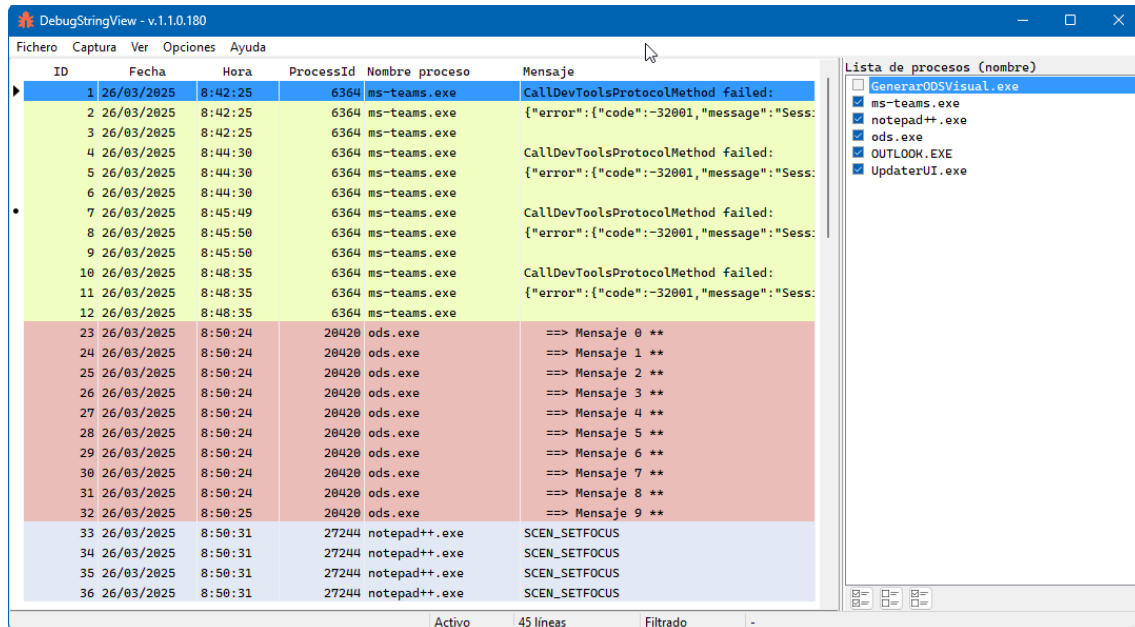

Features

- Allows saving captured content to files with the (*.dsvlog) extension.
- Fast message filtering by process (using the process name), including N processes with different ProcessIDs, but the same name.
- Option to color message output by process (for better and clearer visualization).
- Hide and show columns as needed (more information or more space)
- Automatic scrolling as messages appear, so that the latest messages sent to the console are always displayed.
- 'Process List' panel showing the processes (ProcessName) that are sending messages to the console.
- Allows quick switching of the font used in the display to the most common monospaced fonts. It also provides the option to access the properties screen and change other features.
- The application is multilingual and easily expandable with more language files without requiring changes to the application (dynamic language loading).
- Update checking available.
- Supports bookmark management to mark important positions.
- Quick text search for messages.
- Allows loading log files previously saved by the application.

# Menu Options

The application menu has the following options.

File Menu

- **New capture:** Start a new capture. Current data will be lost.
- **Open**: It will allow you to open a file saved by the same application.
- **Save**: Allows saving the messages currently on screen to a file.
- **Save como**: Allows saving the messages on screen to a new file with a new name.
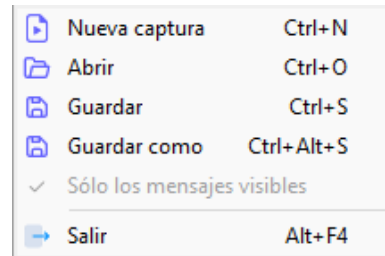
Capture Menu

- **Pause**: Temporarily stops message capture.
- **Resume**: Allows the application to resume capturing debug messages.
- **Clear debug**: Deletes all messages captured by the application so far
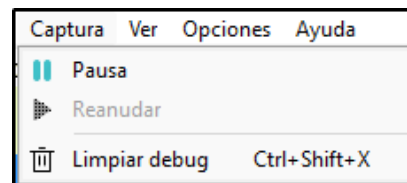
View Menu

- **Columns**: Allows showing and hiding certain columns in the process message view (disabled columns cannot be hidden)
- **Autoscroll**: Automatically advances the message view as new ones appear.
- **Colors by process**: It automatically assigns a different color to each process that appears in the application, to improve on-screen visualization.
- **Process list**: It allows you to view the list of processes that are displaying messages, and the list itself allows you to **filter** messages according to existing processes.
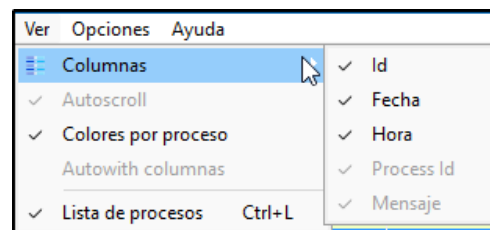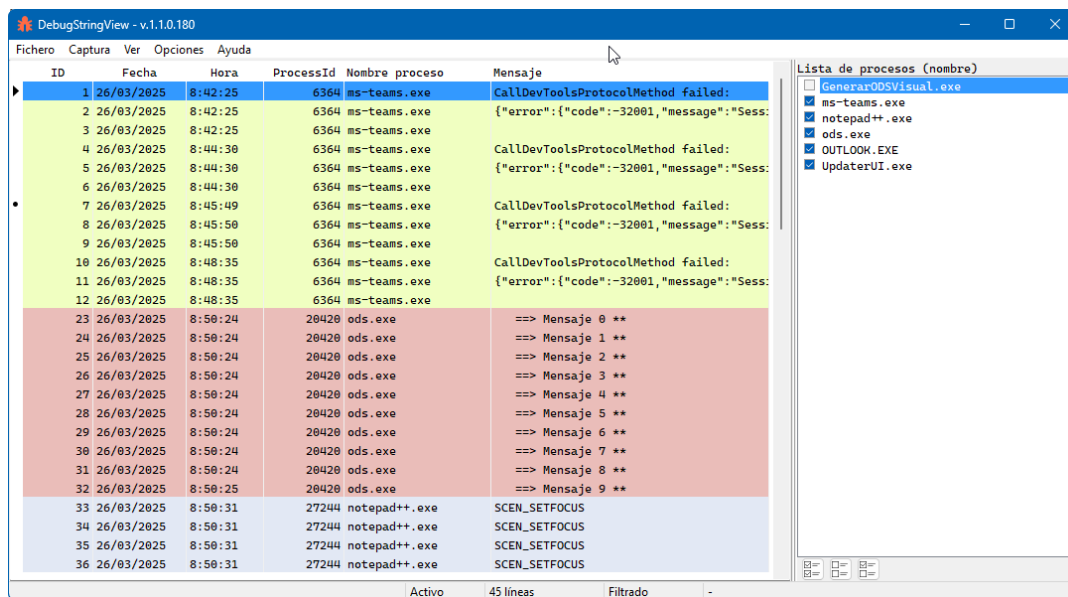
Options Menu

Fonts: Allows you to quickly change the font used to display messages. It also provides an option to access the other properties of the currently used font.
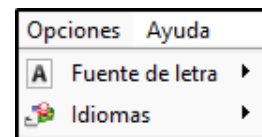
Languages: Allows you to switch between the different languages configured in the application.

Multi-language

The application works with multiple languages, which are also loaded dynamically. More languages can be easily added (without recompiling) and the application will automatically recognize and add them to the menu once it is running.

The application searches the "Languages" folder for language files.

How to generate a new translation

To create a new translation, you must use the free dkLang translation editor/generator (the translation package used in this application).

You can find information about the "DKLang Translation Editor" here:

https://yktoo.com/en/software/dklang-translation-editor/

You can doewnload from here.

To generate a new translation, we only need the file with the original strings **<DebugStringView.dklang>** located in the **<Languages>** directory of the application.

Steps to Create a New Translation

STEP 1: Select the file with the original strings <DebugStringView.dklang>; Additionally, if it is more convenient, you can choose an existing language file in the application as the original file to translate.

<u>Option 1:</u> If we want to generate a new translation (Italian - Italy) from the original translation file **<DebugStringView.dklang>** in Spanish:
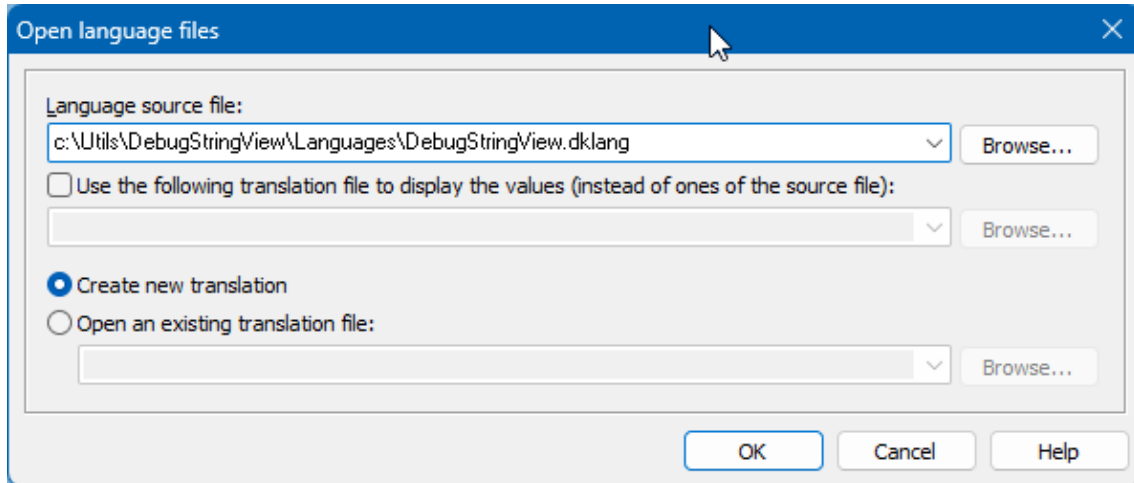
<u>Option 2:</u> We want to generate a new translation (Italian - Italy) using an existing translation (for example, in English).
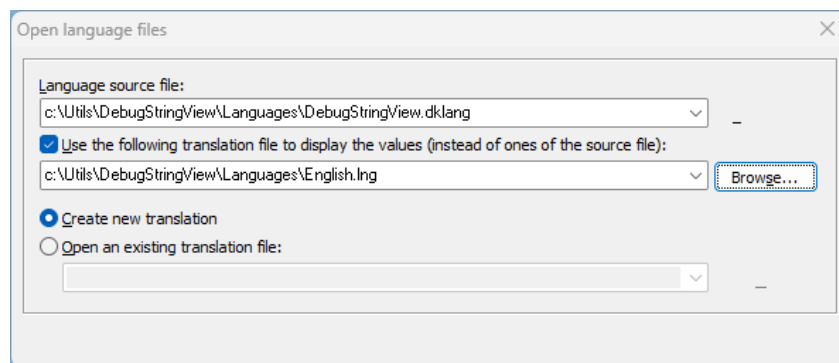
STEP 2: choose the language of the new translation (in this example Italian) and fill in the translation properties.

- **Source Language:** Must be 03082 – Spanish (Spain, International)
- **Translation Language:** Language of the translation; in our case (Italian)
- Other properties: Fill in as desired.

STEP 3: Perform the translations using the editor. The strings on the left are the originals, those on the right are the new translations.

STEP 4: Save the translation file in the application's <Languages> folder. The name should be the name of the new language and the extension should be **.lng**
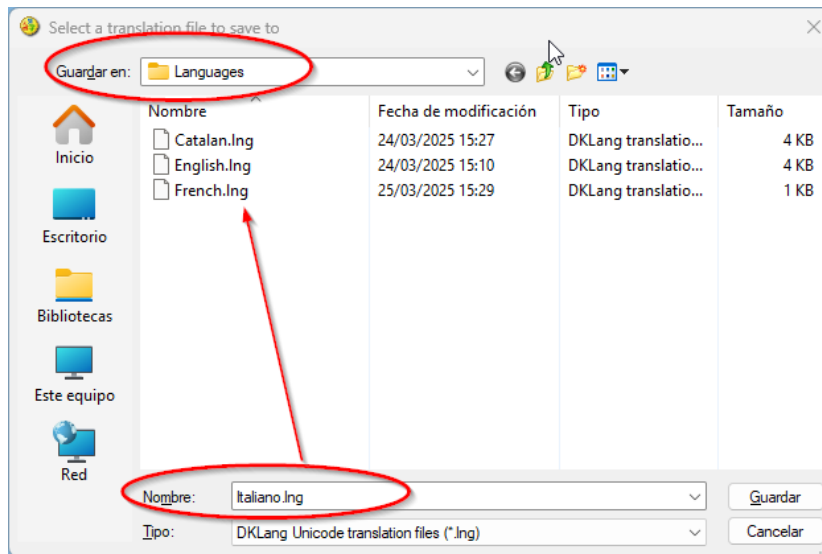
When you reopen the application, the new translation will appear in the **Languages menu**.
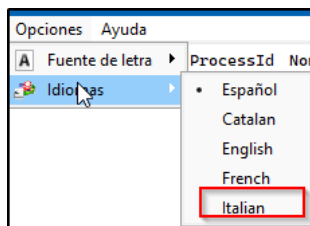
IMPORTANT NOTE: If you create a new translation (in any way), respect the string syntax when you find %s (keep it in the same place in the translated string).